IN THE UNITED STATES PATENT AND TRADEMARK OFFICE

In re patent application of:
   Biplav Srivastava

Serial No.: 10/729,813

Filed: December 5, 2003

Group Art Unit: 2164

Examiner: Pannala, Sathyanaraya R.

Atty. Docket No.: JP920030178US1

For: AUTOMATED INTERPRETATION OF CODES

Commissioner of Patents
P.O. BOX 1450
Alexandria, VA 22313-1450

DECLARATION UNDER 37 C.F.R. §1.131

[0001] I, the inventor of the invention defined by claims 1-15 of U.S. Patent Application Serial No. 10/729,813 hereby declare the following:

[0002] The purpose of this declaration is to prove that I conceived the claimed invention prior to the earliest effective prior art date of U.S. Publication No. 2005/0091640 published to McCollum et al., which is presently understood to be October 24, 2003. The following shows that I conceived my invention prior to October 24, 2003 and that I was diligent from my date of conception to its reduction to practice and was further diligent to the date of the filing of my patent application, which has a filing date of December 5, 2003 (hereinafter referred to as the "Patent Application").

1

[0003] I am the sole inventor of the subject matter claimed in claims 1-15 of U.S. Patent Application Serial No. 10/729,813.

[0004] During all time periods mentioned herein, and specifically between my conception date and the filing date of the application, all activities described herein occurred in India, a WTO Member Country.

[0005] Proof of the conception of the claimed invention prior to October 24, 2003, and diligence in reducing the invention to practice and filing the Patent Application is demonstrated in the attached Exhibits, labeled as Exhibits A and B.

[0006] As shown in Exhibit A, which is an invention disclosure write-up for my invention, I conceived the claimed invention at a date prior to October 24, 2003. As permitted by MPEP §715.07, the dates on Exhibit A have been removed; however, I hereby declare that all dates corresponding to the conception date and reduction to practice occurred prior to October 24, 2003. Further, the invention was actually conceived before Exhibit A was prepared. Therefore, my conception date actually predates Exhibit A.

[0007] Exhibit A specifically discloses in Sections 3-6 the claimed invention as defined by the claims. In fact, the descriptions provided in pages 1-7 of Exhibit A served as the basis for the specification, drawings, and claims of the Patent Application.

[0008] As shown in Exhibit B, which is a Microsoft® PowerPoint Presentation®, I conceived the claimed invention at a date prior to October 24, 2003. As permitted by MPEP §715.07, the dates on Exhibit B have been removed; however, I hereby declare that all dates corresponding to the conception date and reduction to practice occurred prior to October 24, 2003. Further, the invention was actually conceived before Exhibit B was prepared. Therefore, my conception date actually predates Exhibit B.

[0009] Exhibit B specifically discloses on Slides 1-13 the claimed invention as defined by the claims. In fact, the features provided on Slides 1-13 of Exhibit B served as the basis for the specification, drawings, and claims of the Patent Application.

[0010] I was diligent from the date of conception in reducing the invention to practice and in pursuing, preparing, and filing the Patent Application. More specifically, on or about July 30, 2003, information similar to that shown in Exhibits A and B were presented to a patent attorney to determine whether a patent application should be prepared.

[0011] Generally, the invention was conceived on or about March 5, 2003 and was reduced to practice on or about March 30, 2003. An exhaustive series of experiments were conducted on the invention testing its validity from March 5, 2003 to June 30, 2003. The testing was quite rigorous and required substantial time, money, and effort to undertake. The results of the experiments were positive, which further resolved the decision to seek patent protection. After the invention was conceived and reduced to practice, and the testing yielded positive results, the decision was reached to seek patent protection due to the potential commercial value

and prestige afforded by the claimed invention as well as the results of a prior art search. On or about July 30, 2003, a patent attorney was instructed to prepare a patent application that eventually became the Patent Application. The Patent Application was eventually prepared and filed on December 5, 2003.

[0012] The foregoing declarations are made according to my best recollection upon review of the appropriate documents and notes, and I hereby acknowledge that willful false statements and the like are punishable by fine or imprisonment, or both (18 USC §1001) and may jeopardize the validity of the application or any patent issuing thereon. All statements made herein are made of my own knowledge and are true and all statements that are made on information and belief are believed to be true.

Biplav Srivastava

_____
Biplav Srivastava

July 25, 2006

_____
Date

EXHIBIT A

A Method and System for Interpreting Applicable Provisions of "Legally Binding Code" based on User Perspective

Biplav Srivastava

Biplav Srivastava

████████

---

1. Problem:

In the real world, one encounters many types of *legally* binding codes. The term "code(s)" is used to refer to any set of formalized statements of conduct that individuals or entities have to adhere to. These may be laws like laws of natural and criminal justice, international laws; policies, contracts and agreements, etc. We will use the term "law" interchangeably with code but the discussion is valid for the general case of any form of codes.

For any set of codes (e.g., laws), there are many parties with stakes in how the codes are legislated, used for arguments during implementation, prosecution, defense, arbitration, etc. The stakeholders have peculiar perspectives and they want to quickly know what provisions of the code set (e.g., sections of the law) are applicable for them.

This invention provides a system and method that automatically *interprets and recommends the applicable provisions of the code from the code-set based on the user's (stake holder's) perspective to maximize their expected returns.*

The invention has a broad impact. At the micro level, it can help an individual analyze the terms of agreements of a software package quickly before proceeding ahead with its installation. Currently, an average user hardly understands the legalese. At the macro level, national legislatures and legal policy groups can use it to analyze their legal system and detect existing loopholes.

2. Prior Art:

Information Technology (IT) has been used to store and access codes in their many manifestation. For laws, many organizations store them in databases and provide a query interface to search or browse them. However, these searches are based on keywords and do not take the user's goal perspective into account – e.g., the results are the same irrespective of whether the user is a lawmaker, prosecutor or from defense. They do not leverage the inherent structure of codes to facilitate any automated analysis. All analysis is done my humans.

In the case of policies and contracts, humans (usually legal teams or from administration) build code sets and they are manually scrutinized for any validation, violation or analysis.

There does not exist any automatic method for analyzing legally binding codes.

3. Illustrated Example

3.1 Example from Judiciary

For a legal system, the legislature frames (the provisions of) laws, police and enforcement agencies use them to detect legal violations, prosecution relies on them to maximize sentences for accused, defense uses them to ensure fairness in trial, and the judge interprets laws to pass strictures and judgments. However, the judicial process is complicated by the fact that laws have sub-sections, exceptions, amendments, exceptions and finally precedents by virtue of which a legal case is built.

Not all parties who are stakeholders in public law and order are equally conversant with the in-depth provisions of the law. But the fact is that when a crime occurs, the police are asked to file an initial report, which is used as the basis for further investigation. If a mistake is made at any step, it can be used as a legal excuse to insulate the accused.

As a concrete example, in the Indian Penal Code system (IPC), there is Sections 299 to 309 that become applicable upon the suspicious death of a person ranging from homicide, murder, suicide, etc. They are shown in Figure 1.

```
•299. Culpable homicide
•300. Murder
•301. Culpable homicide by causing death of person other than person whose death was intended
•302. Punishment for murder
•303. Punishment for murder by life-convict
•304. Punishment for culpable homicide not amounting to murder
–120[304A. Causing death by negligence
–149[304B. Dowry death
•305. Abetment of suicide of child or insane person
•306. Abetment of suicide
•307. Attempt to murder
•308. Attempt to commit culpable homicide
•309. Attempt to commit suicide
```

Figure1: Sections of IPC about human death

The invention will help the different stakeholders in *best* interpreting the provisions of the penal code with respect to the user's utility perspective:

- Police in not making technical mistakes, legally-speaking
- Prosecutor in drafting the prosecution's case

- Defense in preparing brief for:
    o Post-crime: case disposal, acquittal, minimum sentence
    o Anticipatory defense/ precaution
- Lawmakers in analyzing loopholes and drafting new legislation

It is interesting to note that the system may be implemented in different ways to meet the runtime constraints of the different stakeholders. This is possible by interpreting the law at different levels of abstraction. For example, the police may be interested in a quick interpretation and the system may be accessed (or implemented) on their Personal Device Assistant (PDA). The lawmakers, on the other hand, do not have such hard time constraints but may require that the loopholes be analyzed in depth. Hence, the system could be implemented on powerful servers.

4 Method & System:

We outline the solution as a general template and then discuss how each step can be implemented in different ways. Various embodiments are possible in this space of choices.

4.1 General Solution Template

The key idea in the invention is to represent the codes as a set of rules. Given the rules and optionally, an event (trigger, e.g., crime), which may make the rules applicable, one can evaluate the rules taking the user's perspective into account. This is illustrated in Figure 2.
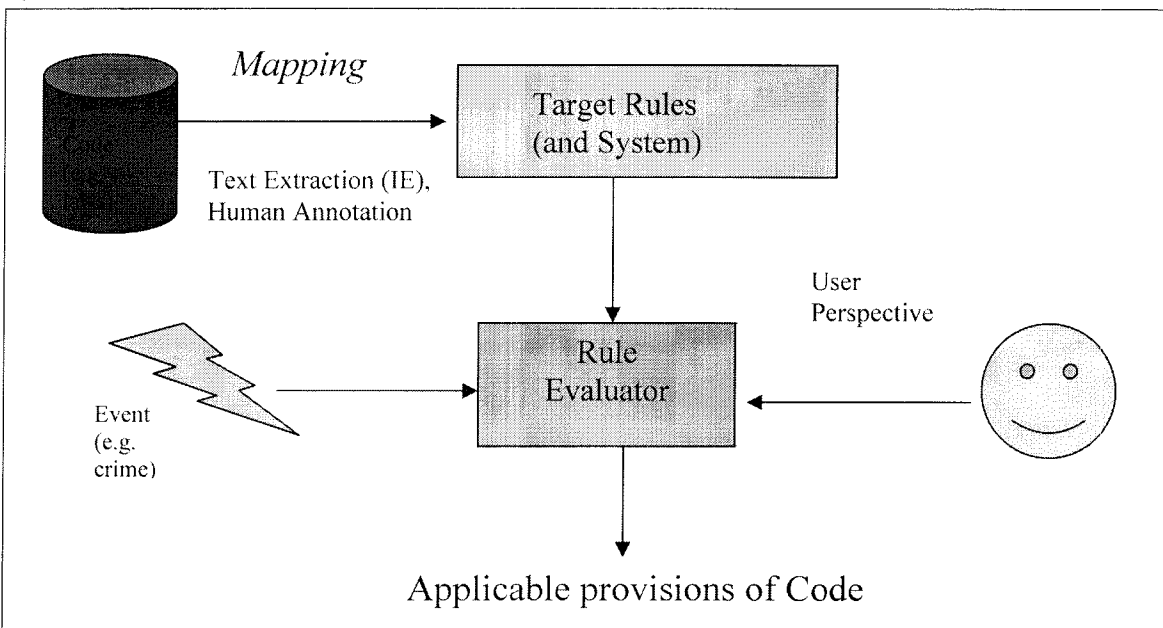

Figure 2: Solution Template

Here is the procedure to implement an automatic code interpretation system.

Step 1: Select a target rule technology

At the outset, a decision is made on the representation of rules and what rule inference system will be used. Some choices for the rules are:

> ➤ Fuzzy rules
> ➤ If-then-else rules
> ➤ Declarative (e.g., prolog) rules

Given the rules, a rule system has to be selected based on performance considerations. A few of them are the desired usage requirements for automatic interpretation in a particular user's perspective (acceptable response time, level of abstraction), hardware available for execution (server, PDA, etc) and overall costs.

Step 2: Select a code to rule transformation technology

The codes have to be mapped to a representation in the selected rules system of Step 1. For this, the representation, structure and complexity of codes have to be modeled/ understood. Next, the each code in the code set has to be translated to the structural template of the target rules (e.g., if-then-else template) and additionally, other input data may be assigned (e.g., fuzzy values for each fuzzy rule).

There are many alternatives to transform the codes into rules. It can be manually performed done by experts who understand both the codes and the rules. Alternatively, standard automated methods from Information Extraction (IE [IE]) and Machine Learning can be used to learn the representational templates of the code and then, the extracted text can be used to populate the target rule templates.

Step 3: Evaluation of rules in the presence of events

Given the code in a machine-readable form (i.e., as rules), we can analyze them based on the user's perspective. Optionally, we may have the information of an event (e.g., crime) that may make some rather than all the rules applicable for analysis. The perspective of the user of the system, depending on their stake vis-à-vis the codes, is represented into an appropriate rule evaluation measure.

Some examples of evaluation functions are:

> ➤ For prosecutor,
>   o $\text{Max } \Sigma\, R_i$ The number of applicable codes be maximum
>   o $\text{Max } \Sigma\, P_{Ri}$ The punishment in the applicable codes be maximum
> ➤ For defense,
>   o $\text{Min } \Sigma\, R_i$ The number of applicable codes be minimum
>   o $\text{Min } \Sigma\, P_{Ri}$ The punishment in the applicable codes be minimum

- For legislature,
  - For any event E, R_i ! empty   The codes are defined for every crime
  - For any event E, P_{Ri} ! 0   No crime goes unpunished

The system recommends rules, and correspondingly the codes, which (expectedly) maximize the user's utility.

5 Relationship to Other Technologies:

5.1 Relationship to Search

The invention is related to search methods (like Google). A search method has following characteristics:

- Keywords are used to store and retrieve documents. Some semantic search (e.g., TREVI in IBM) are appearing that retrieve documents based on additional text annotations. However, they do not leverage the inherent structure of the document - codes in our case.
- It may store rules to specify how to filter results. Specifically, rules are evaluated internally to rank the results. However, there is not much choice for their replacement based on user's utility function. The same result is given to different types of users.
- It is a single, server side implementation that can be accessed by multiple clients.

The key differences between search and the invention are:

- Event may or may not be keyword based
- Rule representation and evaluation modules are replaceable
- Implementation will be done on many types of computing models
- Since the document belongs to a specific domain, i.e., legally binding codes, their cause and effect style structure will be leveraged.

5.2 Relationship to Data Warehousing

There exists a weak relationship between the invention and the data warehousing methods. In the latter, data is aggregated in a repository and operations are provided to view the contents at high-level (abstracted) and low-level (detailed). The main differences are:
- The representation model of data warehouses (e.g., relational) may not be appropriate for translating codes, which are better represented as rules.
- Rule evaluation based on utility do not map to conventional data warehousing operations of aggregating data values

.

5.3 Relationship of code with Business Processes

A business process is a behavioral description of an entity modeled over some time duration. Code on the other hand is the behavior of an entity at a given time instant (e.g., at the time of a crime) and is therefore, a static description.

The invention is applicable for all scenarios where the behavior of the actors can be represented as codes.

6 Impact:

The invention has immense business value in promoting the sale of

●Hardware
–Databases
–Servers (thin clients)
–Pervasive devices (thick clients)

●Software
–Rule solutions: IBM technologies of Policy-based Database Management (PDDA) and Agent Building and Learning Environment (ABLE [ABLE])
–Knowledge Management Technologies (Information Extraction, Machine Translation, NLP) like Web Fountain [WF]

●Consulting Services

The example of market is creates are:

●E-governance:
–Judiciary, law offices
–Law makers
–Individuals

●Corporations:
–Legal cells (contracts, corporate laws),
–Company policy makers for business policies

●Individuals:
–Evaluation of software terms
–Evaluation of tax provisions during filing

■

7 Claims

Broad claims are possible in representing any code (laws, business policies, agreements) as rules and interpreting them automatically based on the user's perspective.

8 References:

1. [ABLE] Agent Building and Learning Environment. http://ableinfo.rchland.ibm.com/
2. [IE] Information Extraction.
   http://www.dcs.shef.ac.uk/research/groups/nlp/extraction/
3. [WF] Web Fountain. http://www.almaden.ibm.com/WebFountain/

EXHIBIT B

A Method and System for Interpreting Applicable Provisions of "Legally Binding Code" based on User Perspective

Biplav Srivastava

Discussed with:
Anupam Saronwala, Ravi Kothari, Mukesh Mohania

Motivation

- Many domains with *legally* binding codes
  - Laws: Laws of justice, International laws, Natural laws
  - Policies: Business policies, Public policies
  - Contracts/ agreements: Non-disclosure agreements (NDAs), Business contracts

- Multiple stake-holders/ users
  - Code makers, prosecutors, arbitrators, defense

- Problem:

  How to interpret and decide the applicable provisions of the code based on user's utility perspective to maximize the goals/returns ?

Example from Judiciary

In Indian Penal Code system (IPC)

- Intricate sections for a crime
  - 299 to 309 in IPC on death (homicide, murder, ...)
  - Many sub-sections, amendments, exceptions and precedents

- Multiple perspectives with different utility functions
  - Prosecutor wants prosecution
  - Defense wants:
    - Post-crime: case disposal, acquittal, minimum sentence
    - Anticipatory defense/ precaution
  - Lawmakers want to analyze loopholes and draft new legislation

- Issue: How to *best* interpret the provisions of the penal code w.r.t. user's utility perspective ?
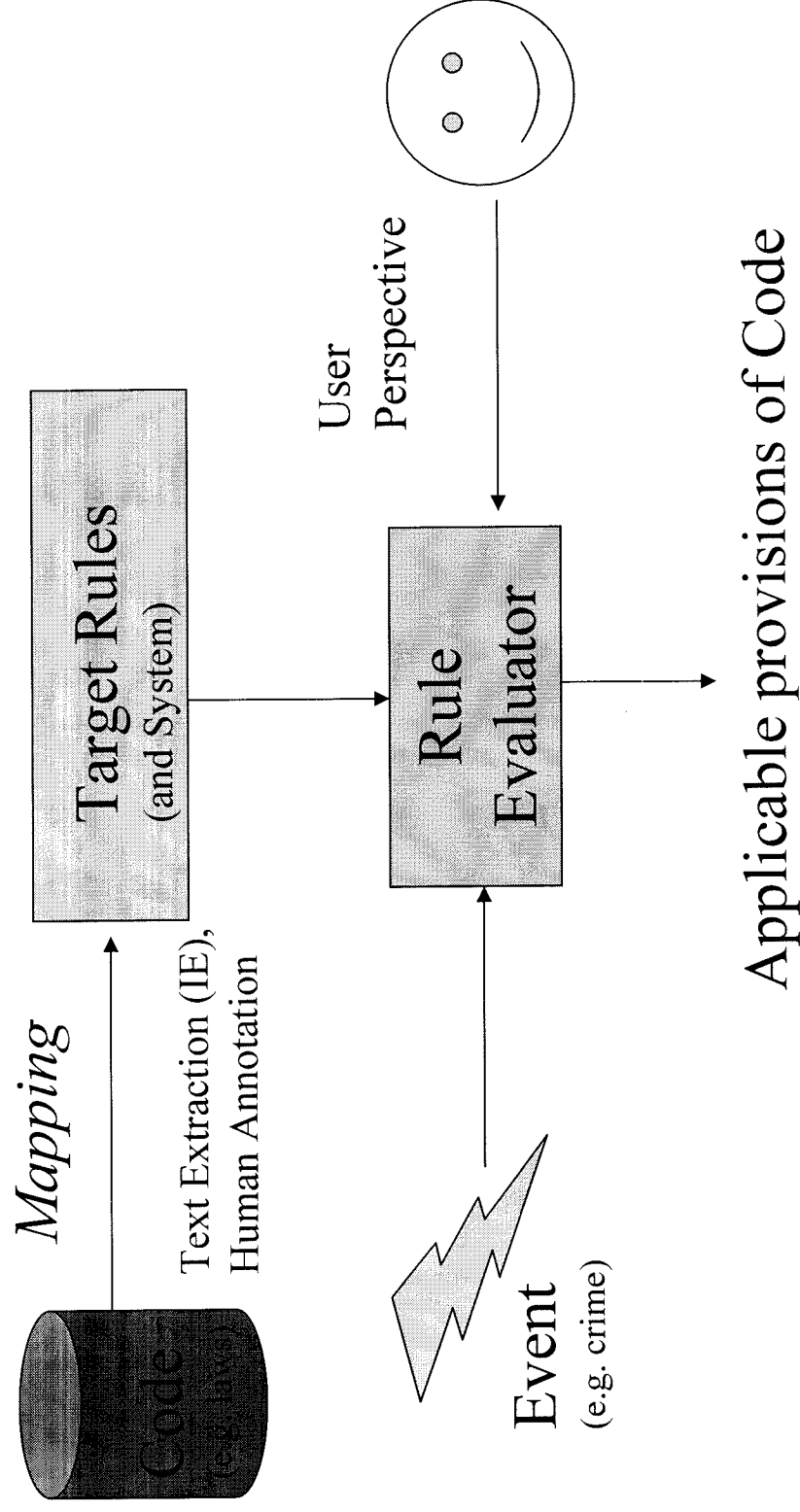
IPC Provisions Relevant to Death

- 299. Culpable homicide
- 300. Murder
- 301. Culpable homicide by causing death of person other than person whose death was intended
- 302. Punishment for murder
- 303. Punishment for murder by life-convict
- 304. Punishment for culpable homicide not amounting to murder
  - 120[304A. Causing death by negligence
  - 149[304B. Dowry death
- 305. Abetment of suicide of child or insane person
- 306. Abetment of suicide
- 307. Attempt to murder
- 308. Attempt to commit culpable homicide
- 309. Attempt to commit suicide

Approach Outline for Judiciary

- Express the laws in the penal system as rules in some representation

- Associate utilities that capture perspectives of different categories of users with different rule evaluation functions

- Given a crime scenario and user perspective, evaluate the rules and return top matches

Assumption: Laws can be represented by rules

Solution Approach in General

Target Rules
(and System)

Mapping

Text Extraction (IE),
Human Annotation

Code
(e.g. laws)

User
Perspective

Rule
Evaluator

Event
(e.g. crime)

Applicable provisions of Code

Issues to Research

- Selection of a target rule technology based on
  - Usage requirements for automatic interpretation in a particular perspective (acceptable response time, level of abstraction)
  - System available for execution (server, PDA,etc)
- Code to rule transformation
  - Representation, structure and complexity of code that needs to be modeled
  - The structure of target rules
  - Translation technology
- Rule Evaluation
  - Mapping user's utility perspective (about <u>code and events</u>) into appropriate evaluation measures

Solution Template

- Selection of a target rule technology
  - Fuzzy rules
  - If-then-else rules
  - Declarative (e.g., prolog) rules
- Code to rule transformation technology
  - Manual
  - Learning methods
- Rule evaluation
  - Utility function defined on rules
  - Recommend rules that (expectedly) maximize utility

Prior Art: Relationship to Search

- Characteristics of a search method

    - Keyword based

    - Combines rule representation and rule evaluation internally; not much choice for their replacement based on user's utility function

    - Usually server side implementation

    - Note: semantic search (e.g., in TREVI) tries to bridge the gap but does not leverage the inherent structure of the document

- Differences

    - Event may or may not be keyword based

    - Rule representation and evaluation modules are replaceable

    - Implementation on many types of computing models

    - Document belongs to a specific domain, legally binding codes. Hence, have structure that can be leveraged.

Prior Art: Relationship to Warehousing

- Characteristic of a data warehouse
  - Any type of data can be stored
  - Stores data in a determined representation model
  - Allows high-level (abstracted) and low-level (detailed) view of the content (rollup/drill down)

- Differences
  - The representation model (e.g., relational) may not be appropriate for translating codes
  - Rule evaluation based on utility may not map to conventional data warehousing operations

Code v/s Business Process

- Characteristics of a business process
  - A dynamic actor (or entity) whose behavior is modeled over a time duration

- Characteristic of a code
  - A static specification of behavior of an actor (behavior) at a given time instant.

The invention is applicable for all scenarios where the behavior of the actors can be represented as codes.

Immense Business Value

- Hardware
  - Databases
  - Servers (thin clients)
  - Pervasive devices (thick clients)
- Software
  - Rule solutions: PDDA, ABLE
  - Knowledge Management Technologies (Information Extraction, Machine Translation, NLP)
- Consulting

Examples of Markets

- E-governance:
  - judiciary, law offices
  - law makers
  - individuals

- Corporations:
  - legal cells (contracts, corporate laws),
  - company policy makers

- Individuals:
  - evaluation of software terms